What is claimed is:

1. A small footprint device comprising:

a. at least one processing element;

b. memory,

c. a context barrier for isolating one program module from at least one other program module using said memory and processing element, and

d. an entry point object for permitting one program module to access one other program module across said context barrier.

2. The small footprint device of claim 1 in which said one other program module is a supercontext of said one program module.

3. The small footprint device of claim 2 in which said entry point object can access program modules regardless of the memory space in which they are located.

4. The small footprint device of claim 1 in which said entry point object contains an access method having privileges to access at least said another

program module.

5. The small footprint device of claim 1 in which said entry point object calls an object to be accessed in said other program module and switches context to said other program module.

6. The small footprint device of claim 5 in which said other program module performs access to said object to be accessed and returns the results of said call to said one program module.

7. The small footprint device of claim 1 in which said context barrier allocates separate name spaces for each program module.

8. The small footprint device of claim 1 in which said context barrier allocates separate memory spaces for each program module.

9. The small footprint device of claim 1 in which said processing element is single threaded.

10. The small footprint device of claim 9 in

which the processing element runs each program module as a separate context.

11. The small footprint device of claim 10 in which said entry point object issues a call to a program module containing an object to be accessed and then causes said processing element to switch context to the programming module containing said object to be accessed.

12. The small footprint device of claim 1 in which at least one program module comprises a plurality of applets.

13. The small footprint device of claim 1 in which said context barrier enforces security checks on at least one of a principal, an object and an action.

14. The small footprint device of claim 13 in which at least one security check is based on partial name agreement between a principal and an object.

15. The small footprint device of claim 14 in which said entry point object can access objects

without said at least one security check.

16.   The small footprint device of claim 13 in which at least one security check is based on memory space agreement between a principal and an object.

17.   The small footprint device of claim 16 in which said entry point object can access objects without said at least one security check.

18.   A method of operating a small footprint device, comprising the step of separating program modules using a context barrier and permitting access across the context barrier using an entry point object.

19.   The method of claim 18, in which the context barrier is implemented using a single threaded processing element.

20.   The method of claim 18, in which said entry point object accepts requests from one program module for access to an object and then switches context to a program module containing said object where the access can be performed.

21. The method of claim 19 in which the context barrier will not permit a principal to perform an action on an object unless both principal and object are part of the same context and the action is appropriate for the object unless the request comes via an entry point object.

22. A computer program product, comprising:

a. a memory medium; and

b. a computer controlling element comprising instructions for implementing a context barrier on a small footprint device and for bypassing said context barrier using a an entry point object.

23. The computer program product of claim 22 in which said memory medium is a carrier wave.

24. A computer program product, comprising:

a. a memory medium; and

b. a computer controlling element comprising instructions for separating a plurality of programs on a small footprint device by running them in respective contexts and for permitting one program to access

another program using an entry point object.

25. The computer program product of claim 24 in which said memory medium is a carrier wave.

26. The computer program product of claim 24 in which said another program is a programming running as a supercontext of said one program.

27. A carrier wave carrying instructions for implementing an entry point object for bypassing a context barrier on a small footprint device over a communications link.

28. A carrier wave carrying instructions over a communications link for separating a plurality of programs on a small footprint device by running them in respective contexts and for permitting one program to access another program using an entry point object.

29. A method of transmitting code over a network, comprising the step of transmitting a block of code from a server, said block of code comprising instructions for implementing an entry point object for

5   providing access across a context barrier.